SUB C5

A4

18.   (Amended)   A system for managing software conflicts, the system comprising:

means for determining change information which represents actual changes made to a computer system's files and other shared resources during installation of different applications into the computer system;

means for processing the change information to determine conflict information which indicates which files and shared resources conflict with one another;

a database of interrelated tables for storing the conflict information; and

means for resolving the software conflicts based on the stored conflict information.

Remarks

In the Office Action dated January 29, 2001, the Examiner indicated that the drawings previously filed were approved by the draftsperson. The Examiner also required the Applicant to submit PTO Form 1449 with the prior art indicated on pages 3 and 4 of the specification. The Examiner also required a new title. The Examiner rejected claims 1, 5-8, 14-18, 22-25, 31 and 32 under 35 U.S.C. § 102 as being anticipated by the U.S. Patent to Stupek, Jr., et al. 5,586,304. The Examiner rejected claims 2-4 and 19-21 under 35 U.S.C. § 103 as being unpatentable over Stupek, Jr., et al. in view of the U.S. Patent to Shipley 5,634,114. The Examiner rejected claims 9-13 and 26-30 under 35 U.S.C. § 103 as being unpatentable over Stupek, Jr., et al. in view of the U.S. Patent to Choye, et al. 5,842,024.

By this Amendment, Applicants' attorney has submitted PTO Form 1449. Applicants' attorney has also submitted a new title. Finally, Applicants' attorney has amended each of the independent claims to more particularly point out and distinctly claim what Applicants regard as their invention. In particular, each of the independent claims have been amended to make it clear that the present invention determines change information which represents actual changes made to a computer system's files and other shared resources during installation of different applications into the computer system. Also, the two independent

claims requires processing the change information to determine conflict information which indicates which files and shared resources conflict with one another.

Clearly, the above-noted features of the present invention are neither taught, disclosed or discussed by prior art references of record in this application. For example, the U.S. Patent to Stupek, Jr., et al. discloses upgrading computers by comparing version information on a local computer with that of a central data base for the purpose of upgrading a single application on the local computer. The table described by Stupek, Jr., et al. in Figure 5b contains information for the version of packages that can be used to upgrade (via comparing the version information stored in the table against a local computer's version information). Stupek, Jr., et al. does not make reference to populating this table of installation information of applications. In addition, the tables referenced by Stupek, Jr., et al. do not reference the application installation information provided by the present invention. The tables referenced in Figures 5a, 5b and 5c do not reference the changes that will actually be made to the system itself. They only reference version numbers and comparison information to be compared with the local system itself.

This should be contrasted with the present invention wherein information about the actual changes performed by an application installation is determined and placed into tables.

The Examiner further interprets Figure 5b of Stupek, Jr., et al. to be equivalent to change information. However, the information stored in Figure 5b of Stupek, Jr., et al. only describes version, date, description type and importance not changes themselves. There is no information in Figure 5b to describe the list of such resources that will actually be affected by the package itself. Figure 5b only provides a description of the package with no information about the contents of the package itself.

The Examiner further interprets Figure 5c element 29 of Stupek, Jr., et al. to be equivalent to storing conflict information in a data base of interrelated tables . The information stored in this table, however, describes how the version information will be used

to compare to the version of the package on the local computer. The comparison is then used to determine if a package will be installed.

The conflict information of the present invention is not used to determine if a resource will be installed, as in Stupek, Jr., et al. Instead, the conflict information of the present invention indicates if a number of resources are incompatible with each other. The conflict information does not depend on version information at the package level. Rather, it depends on the type of changes that are being made by the contents of the different packages. For example, two registry keys created by different applications may conflict with each other if they try to place a different value into the same registry key. In this case, there is no comparison of version information. The individual resources conflict when the conflict information is stored in the database itself.

None of the other prior art references of record make up for the deficiencies of Stupek, Jr., et al noted above. For example, the U.S. Patent to Shipley merely describes a system wherein a program asks a DLL if it supports a certain version number. The DLL itself is written containing a table of version numbers it supports. The DLL then lets the program know whether or not it supports the version the program is requesting. Shipley's system is a method of programming applications and DLL's.

The U.S. Patent to Choye, et al. merely discloses a software installation method which creates modules for each software program to be installed and by associating a file with each software program that records all changes made to files and directories by the installation process of the software program. However, Choye, et al. does not disclose change and conflict information of the present invention which are utilized to determine and resolve software conflicts.

Consequently, in view of the above and in the absence of better art, Applicants' attorney respectfully submits that the application is in condition for allowance which allowance is respectfully requested.

A check in the amount of $445.00 is enclosed to cover the Petition fee. Please charge any additional fees or credit any overpayments as a result of the filing of this paper to our Deposit Account No. 02-3978 -- a duplicate of this paper is enclosed for that purpose.

Respectfully submitted,

JOHN J. MCMILLAN, ET AL.

By_____
David R. Syrowik
Reg. No. 27,956
Attorney/Agent for Applicant

Date: July 24, 2001

BROOKS & KUSHMAN P.C.
1000 Town Center, 22nd Floor
Southfield, MI 48075
Phone: 248-358-4400
Fax: 248-358-3351

Attachment

VERSION WITH MARKINGS TO SHOW CHANGES MADE

1.   (Amended) A method for managing software conflicts, the method comprising the steps of:

determining <u>change information which represents actual</u> changes made to a computer system's files and other shared resources during installation of [at least one application] <u>different applications</u> into the computer system [to obtain change information];

processing the change information to determine <u>conflict information which indicates</u> which files and shared resources conflict with one another [to obtain conflict information];

storing the conflict information in a database of interrelated tables; and

resolving the software conflicts based on the stored conflict information.

16.   (Amended)   A computer-readable storage medium having stored therein a program which executes the steps of:

determining <u>change information which represents actual</u> changes made to a computer system's files and other shared resources during installation of [at least one application] <u>different applications</u> into the computer system [to obtain change information];

processing the change information to determine <u>conflict information which indicates</u> which files and shared resources conflict with one another [to obtain conflict information];

storing the conflict information in a database of interrelated tables; and

resolving software conflicts based on the stored conflict information.

18.   (Amended)   A system for managing software conflicts, the system comprising:

means for determining <u>change information which represents actual</u> changes made to a computer system's files and other shared resources during installation of [at least one application] <u>different applications</u> into the computer system [to obtain change information];

means for processing the change information to determine <u>conflict information which indicates</u> which files and shared resources conflict with one another [to obtain conflict information];

a database of interrelated tables for storing the conflict information; and

means for resolving the software conflicts based on the stored conflict information.